Process Engineering

A Systems Approach to Process Improvement

Jeffrey L. Dutton

Jacobs Sverdrup Advanced Systems Group

Engineering Performance Improvement Center

A funny thing happened on the way to the briefing…

- Realization that “process engineering” is only part of the answer
  - Puts process development in a systems development context
  - Inferred and supported by the CMMI℠ Model Suite
  - But - omits other functions that are critical to the development of engineering capability
- Decision to integrate “process engineering” into a larger systems context

JACOBS SVERDRUP

Process Engineering

A Systems Approach to Process Improvement

Engineering Systems Development

A Systems Approach to Engineering Capability

JACOBS SVERDRUP

~~Process Engineering~~

~~A Systems Approach to Process Improvement~~

Engineering Systems Development

A Systems Approach to Engineering Capability

Some Reasonable Questions

- What do we mean by an “engineering system”?
- Why would we go to the trouble of looking at engineering capability in this way?
- How do we go about building or improving an engineering system?

Some Reasonable Questions

- What do we mean by an “engineering system”?
- Why would we go to the trouble of looking at engineering capability in this way?
- How do we go about building or improving an engineering system?

What is an Engineering System?

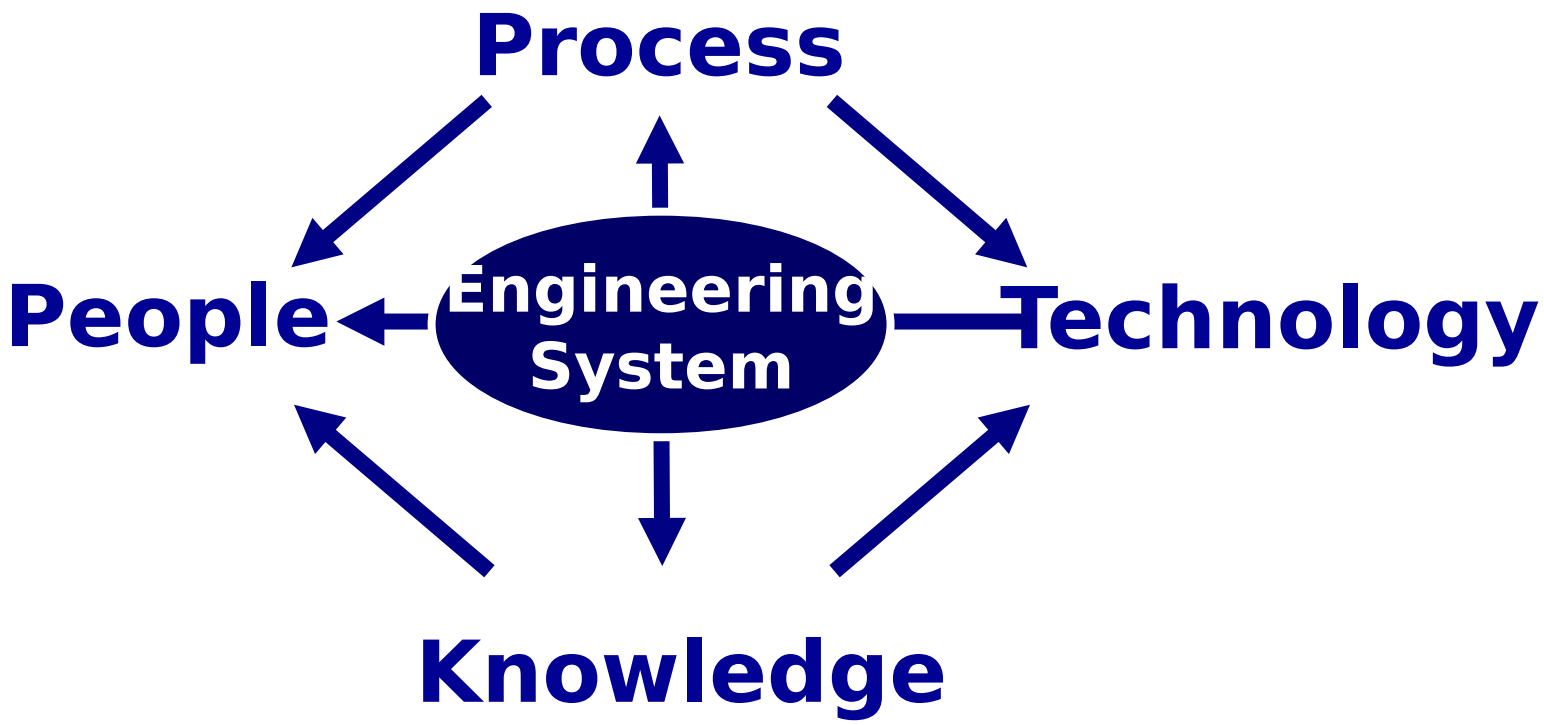
- Engineering system = capability for the development of systems, hardware, or software
- Components:
  - People
  - Process
  - Technology
  - Knowledge
- Interfaces:
  - Internal (among components)
  - External (to stakeholders and customers)

What do these terms mean?

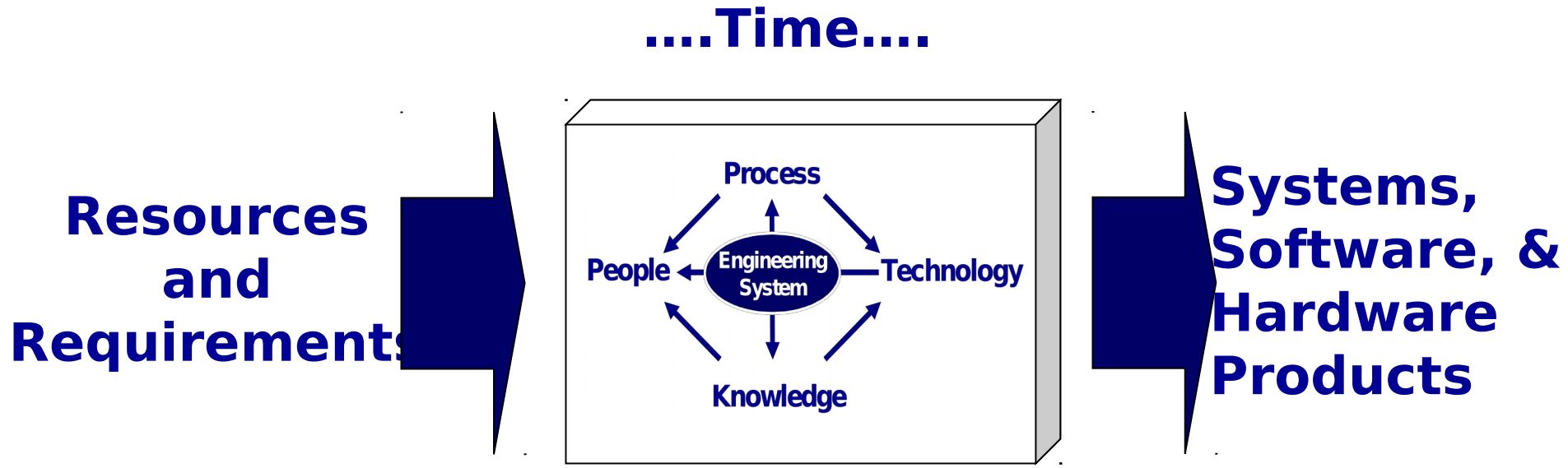
(Components of an engineering system)

- People: the people who are part of the engineering system
  - Engineers
  - Infrastructure support personnel
  - Managers
- Process: the processes used by the people or technology to accomplish the functions of the engineering system
- Technology: the tools and mechanisms of the engineering system
- Knowledge: value-added contextual information necessary to the development and operation of the engineering system

Primary Relationship Diagram



Engineering System Capability



Some Reasonable Questions

- What do we mean by an “engineering system”?
- Why would we go to the trouble of looking at engineering capability in this way?
- How do we go about building or improving an engineering system?

Why go to the trouble?

- Clear requirements leading to clear capability
- Ability to make functional trades
- Ability to clearly define, develop, and manage interfaces
- Ability to define and improve performance of the engineering system
- Knowledge and predictability of what it would take to make the engineering system do something new, different, or better
- Ability to make informed investment decisions

Ability to define and improve performance

(of the engineering system)

- Technical Performance Parameters (system level)
  - Throughput
  - Efficiency
  - Productivity
  - Product defect rates
  - Domain migration ability
  - Maintenance cost
  - Availability and reliability
- Configuration Item level
  - Process capability
  - Technology capability
  - Personnel skills and education
  - Knowledge base
- Interface level
  - Ability to carry information
  - Ability to support functional relationships

Predictability of system changes

- Adopting more rigorous or new performance parameters
- Migrating to new domains (producing something different)
- Taking advantage of new technologies
- Migrating to new processes (like the CMMI℠)
- Increasing capacity of the engineering system
- Integrating engineering systems for software and hardware
- Configuration management of system

Some Reasonable Questions

- Why would we go to the trouble of looking at engineering capability in this way?
- What is different, unique, or value-added in this approach?
- How do we go about building or improving an engineering system?

Engineering System Development

- Elicit customer and stakeholder needs
- Define requirements
- Allocate and validate requirements
- Design engineering system
- Verify and implement design
- Deploy and transition system or system components

Elicitation of Customer and Stakeholder Needs

- External customer and stakeholder needs
  - Domain requirements
  - Product specification performance impacts
  - Product complexity expectations
  - Customer communications requirements
  - Delivery and transition requirements
- Internal customer and stakeholder needs
  - Business policies and rules
  - System performance goals
  - Infrastructure stakeholder needs

Operational Architecture

Engineering Systems requirements analysis

- Functional requirements
  - Program Management capability
  - System, hardware, and software development capability
  - Systems analysis and control capability
  - Domain migration ability
- Performance requirements
  - Throughput, efficiency, and productivity
  - Product defect rate goals
  - Develop system level Technical Performance Measures
- Design constraints
  - CMMI Model Suite requirements
  - Business/ technology domain(s)
  - Project size and complexity (nominal and spread)
  - Business and quality goals
  - State of current engineering culture
  - State of process culture and best practices
- Security and trust requirements
- Availability and Maintainability requirements

Technical Architecture

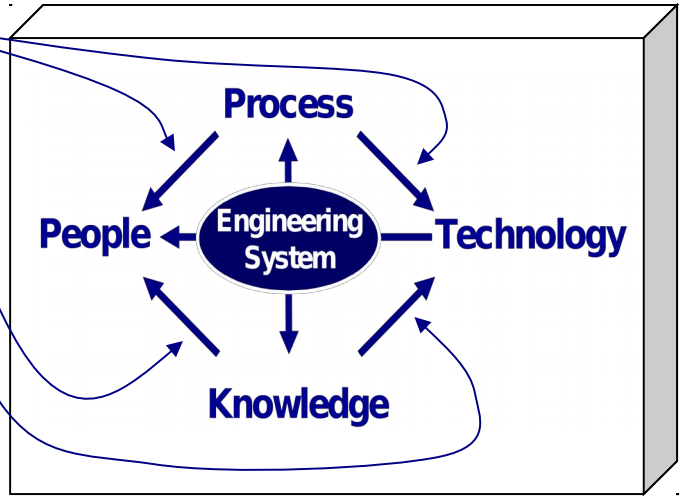
Requirements allocation for an Engineering System

System Level Requirements

Functional Allocation

Internal Interface Requirements

Process Requirements

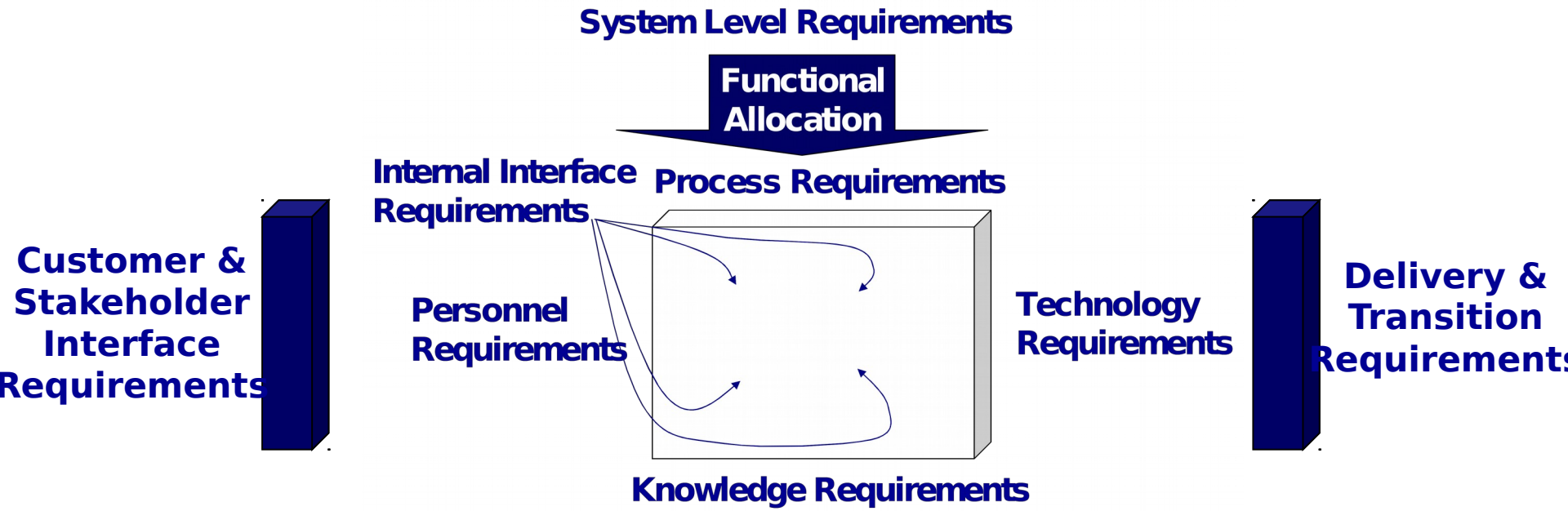


Personnel Requirements

Technology Requirements

Knowledge Requirements

Requirements allocation for an Engineering System



Engineering System Design

- Define System Architecture
- Identify and track system technical performance parameters
- Design components and interfaces in response to allocated requirements
  - Process design
  - Technology design
  - Personnel skills and educational criteria
  - Knowledge system design
  - Interface designs

System Architecture

Verify and Implement Design

- Implement Processes
  - Process implementation may be in technology
- Implement Technology
- Implement Knowledge Assets
  - Build knowledge assets over time
  - Build in strategic phases to match system evolution
- Implement People approach
  - Affect hiring, training, job assignment, and knowledge assimilation
- Implement internal and external interfaces
- Continuously verify design integrity

Deploy Engineering System builds

- Ensure Engineering System baselines are internally congruent (design verification and configuration audit)
- Deploy in increments that provide system level capabilities
- Precede deployment with training and knowledge dissemination
- Practice configuration management of components and baseline control of engineering system

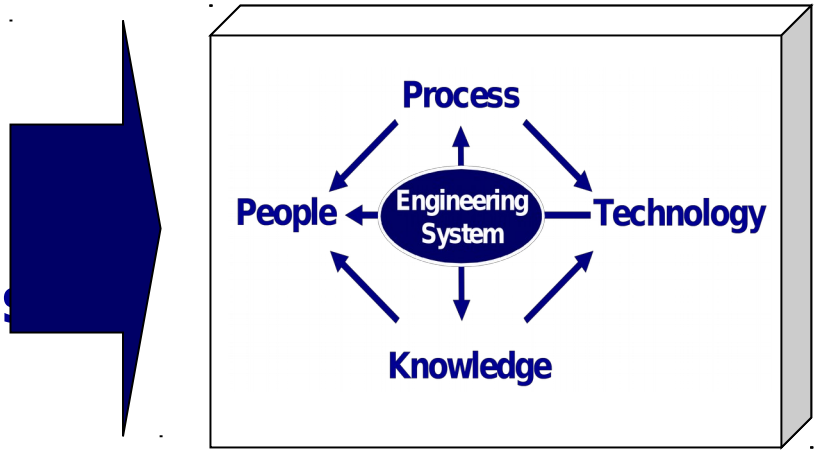
Summary

- Introduced the idea that the capability to develop systems, software, or hardware can be treated as an Engineering System
- Provided summary of the systems life cycle for a notional Engineering System
- The Engineering System approach provides the abilities to:
  - Make functional trades among system components
  - Clearly define, develop, and manage interfaces
  - Define and improve performance of the engineering system
  - Predict what it would take to make the engineering system do something new, different, or better
  - Adopt more rigorous or new performance parameters
  - Migrate to new domains (producing something different)
  - Take advantage of new technologies
  - Migrate to new processes (like the CMMI℠)
  - Increase capacity of the engineering system
  - Integrate engineering systems for software and hardware
  - Apply configuration management principals to engineering capability

Engineering System Capability

....Time....

Resources and Requirements



Systems, Software, & Hardware Products

Questions?

BACK UP SLIDES

Requirements Development Slides

Notional Process Requirements

- Satisfy accepted requirements of CMMI Model(s)
- Support development of products in identified technical domains (life cycle)
- Optimize process for predicted project size
- Accommodate alternative project sizes
- Optimize process for predicted project complexity
- Support accomplishment of business and quality goals
- Support transition of engineering, management, and process cultures
- Integrate current best practices and identify appropriate targets for improvement
- Define People and Technology interfaces
- Define interfaces to business infrastructure

Notional Technology Requirements

- Implement processes in a cost effective manner
- Make knowledge accessible in a timely manner in accordance with Knowledge interface requirements
- Integrate Knowledge into technology where appropriate
- Provide appropriate human interfaces to People
- In C4ISR Terminology, define the Technical Architecture
- Define technical interfaces to business infrastructure

Notional Knowledge Requirements

- Define knowledge domains or assets that support and enable process and technology implementation
- Define knowledge creation requirements for future growth and evolution of the business base
- Provide Knowledge Integration and Knowledge Access requirements to Process, Technology, and People

Notional People Requirements

- Define skill sets that take advantage of Knowledge assets, Technology, and Process
- Define appropriate educational requirements
- Define Knowledge that must be assimilated as a function of job execution
- Define interface requirements for Process, Technology, and Knowledge
- Define interfaces to Human Resources

Design Slides

Notional Process Design

- Design the Process Architecture
  - Primary processes
  - Process relationships
  - Process fidelity
- Define interface to the Technology Architecture
- Define tailoring approach and criteria
- Define structure of standard process(es)
- Define component processes and internal interfaces

Notional Technology Design

- In C4ISR Architecture Framework terminology, implement the Technical Architecture and allocated portion of System Architecture
- Design process implementation
- Design the interface to existing and future business systems
- Design the interfaces to existing and future product design and implementation systems
- Design the interfaces to Knowledge assets
- Design the interfaces to People

Notional Design of Knowledge Assets

- Design the Knowledge Architecture
- Design knowledge gathering and creation functions
- Design knowledge assimilation functions
- Design knowledge assessment functions
- Design interfaces to Process assets, Technology base, and People

Notional Design of People Assets

- Match target skills and education to job functions
- Match job functions to processes, technology, and knowledge assets
- Design interfaces to processes, technology, and knowledge assets
- Design interfaces to Human Resources functions
- Design risk mitigation approach to mis-match of skill and educational sets with job functions